2024中国智能机器人格斗及竞技大赛

智能机器人创新设计大赛

基于VEX系统的未来式物流搬运模块化轻量化

智能机器人

设计报告

队伍名称：TJVEXA

所属单位：同济大学

电子与信息工程学院

团队成员：谷大森

李帅嘉

朱俊泽

杨首望

目 录

[1 项目背景及设计原因 - 1 -](#_Toc180788522)

[1.1 项目背景以及发展状况 - 1 -](#_Toc180788523)

[1.2 作品设计原因 - 1 -](#_Toc180788524)

[2 竞技规则简介 - 3 -](#_Toc180788525)

[3 机械结构部分创新 - 5 -](#_Toc180788526)

[3.1 结构设计 - 5 -](#_Toc180788527)

[3.1.1 底盘设计 - 5 -](#_Toc180788528)

[3.1.2 上层结构设计 - 6 -](#_Toc180788529)

[3.2 材料选择 - 8 -](#_Toc180788530)

[4 格斗机器人AI、程序创新 - 10 -](#_Toc180788531)

[4.1 程序 - 10 -](#_Toc180788532)

[4.1.1 底盘控制 - 10 -](#_Toc180788533)

[4.1.2 功能函数 - 11 -](#_Toc180788534)

[4.2 AI应用 - 12 -](#_Toc180788535)

[4.2.1 基本detection - 12 -](#_Toc180788536)

[4.2.2 yolo - 12 -](#_Toc180788537)

[4.2.3 detection的python部署 - 13 -](#_Toc180788538)

[4.2.4 数据处理 - 14 -](#_Toc180788539)

[4.2.5 检测可视化 - 16 -](#_Toc180788540)

[4.2.6 主控通信 - 17 -](#_Toc180788541)

[4.2.7 决策 - 19 -](#_Toc180788542)

[4.2.8 应用 - 25 -](#_Toc180788543)

[5 总结与未来展望 - 26 -](#_Toc180788544)

[5.1 总结 - 26 -](#_Toc180788545)

[5.2 未来展望 - 26 -](#_Toc180788546)

# 1 项目背景及设计原因

## 1.1 项目背景以及发展状况

近年来，机器人和人工智能领域取得了迅猛的发展，推动了多个行业的技术革新。机器人技术逐渐从制造业延伸到服务业、医疗、农业等领域，自动化生产和智能操作大幅提高了工作效率，降低了成本。例如，在制造业中，工业机器人广泛应用于装配、焊接和搬运等高重复性任务，而在医疗领域，手术机器人通过高精度的操作协助医生完成复杂的外科手术。

人工智能具有技术属性和社会属性高度融合的特点，是经济发展新引擎、社会发展加速器。大数据驱动的视觉分析、自然语言理解和语音识别等人工智能能力迅速提高，商业智能对话和推荐、自动驾驶、智能穿戴设备、语言翻译、自动导航、新经济预测等正快速进入实用阶段，人工智能技术正在渗透并重构生产、分配、交换、消费等经济活动环节，形成从宏观到微观各领域的智能化新需求、新产品、新技术、新业态，改变人类生活方式甚至社会结构，实现社会生产力的整体跃升。同时，加快人工智能在教育领域的创新应用，利用智能技术支撑人才培养模式的创新、教学方法的改革、教育治理能力的提升，构建智能化、网络化、个性化、终身化的教育体系，是推进教育均衡发展、促进教育公平、提高教育质量的重要手段，是实现教育现代化不可或缺的动力和支撑。

随着5G、云计算和物联网（IoT）等技术的进步，机器人和AI的应用前景更加广阔。通过高速、低延迟的网络连接，AI与机器人的协作效率将进一步提升，实现更复杂的自动化操作。

与此同时，机器人在教育领域的兴起也势如破竹，它正在改变传统的教学模式，提升教育的个性化、互动性和普及性。随着机器人技术的成熟和人工智能的进步，教育机器人不仅能承担辅助教学的任务，还可以作为学生的学习伙伴，帮助培养学生的创造力、解决问题的能力以及科技素养。

不仅如此，“竞技”这个词条也早已在机器人的属性上留在了深深的印记。竞技机器人在整个机器人领域着重表现了机器人在技术水平上的创新，相较于工业机器人、服务机器人而言，竞技机器人的竞技特性更是让其能够具有更多的创新、发展空间，以竞技为手段不断拓展机器人的功能特性，并且加以AI赋能机器人以更多未来化特征。

因此可以说，机器人与AI的融合发展在不远的未来势必大放光彩。

## 1.2 作品设计原因

习近平总书记在党的二十大报告中强调：“加强企业主导的产学研深度融合，强化目标导向，提高科技成果转化和产业化水平。”正如本次大赛介绍所言：“智能机器人已经走进人们的生活,成为人们生活的得力帮手”。不仅仅应当将机器人停留在实验室里的“高大上”层次当中，更应该集合实际将机器人的应用推广到方方面面。

在人工智能与机器人技术高速发展的今天，我们可以切身体会到物流搬运机器人的重要性。但目前而言，多数机器人大多停留于物品的水平运动，例如流水线运动、直线物流搬运小车等工业机器人；同时缺少人工智能精确、快速地识别物品、规划物品。因此我们基于VEX V5控制系统与元件，在竞赛模板上将未来的实物进行**理想化建模**，抽象成为圆环类物品，为未来式物流搬运提供一种新的思路与解决方案，同时也**富有竞技性**。

在这个自动化搬运的设计方面，我们希望实现一个**简单、轻量化、模块化的搬运设计**。VEX本身的机械搭建设计成本相对低廉，为了在AI方面实现这个特点。我们在目标检测的方面设计了**yolo模块**，**部署方便，数据集要求较小**，对训练的成**本要求比较低**。同时在ai决策方面，我们使用了jetson nano b01 基础版作为硬件，**算力刚好满足规划和决策需求而不溢出**。然后在决策的算法上，我们使用**模块化**的算法设计，输入场地信息（检测到的信息），通过改变算法内容实现决策。

并且我们团队希望借助此项目可以填补VEX机器人在教育领域该特定方向下的空白，更希望能为未来的工程领域的应用提供一种可能的解决方案，更是希望能在机器人关于“竞技”方向的创新设计上更进一步。

# 2 竞技规则简介

本智能机器人主要基于VEX控制系统，竞技规则为REC Foundation 旗下的VEX Robotics Competition 机器人竞赛发布的24-25 赛季主题名为High Stake 的竞赛规则。由于规则细则过于复杂，现将其简化后提炼出于竞技得分相关的规则陈列于下：

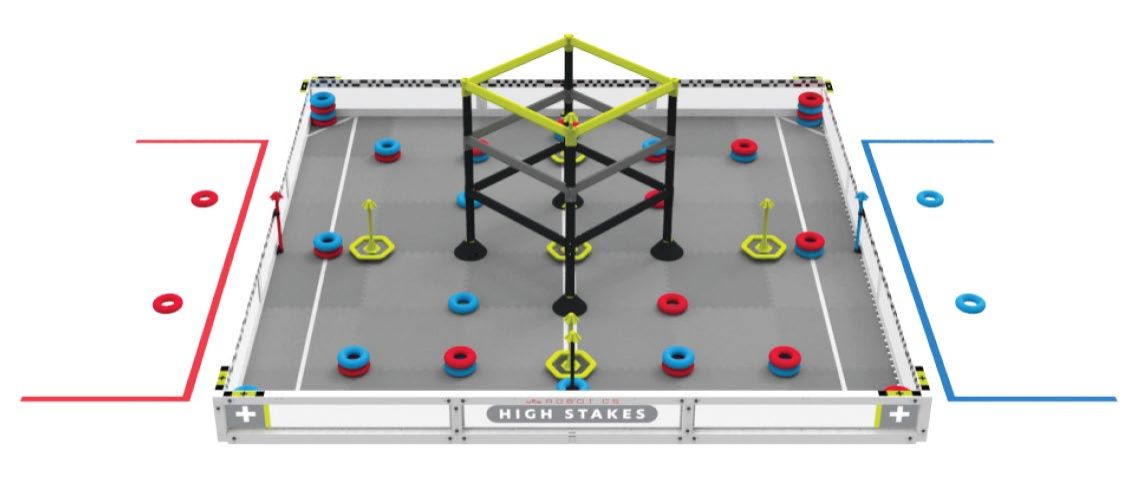
1. V5RC 挑战赛尖峰时刻在 12’ x 12’的正方形场地上进行，如下图和本手册中的其他图示。

图2-1 场地示意图

2. 赛局目标是通过将套环放在尖桩上得分，挪动移动桩，在赛局结束时攀爬，以获得比对方联队更高的得分。

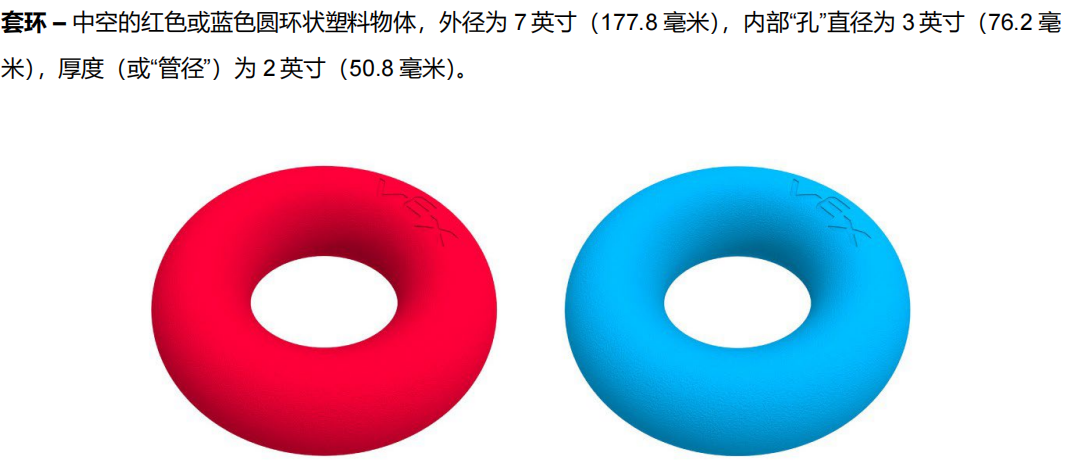
图2-2 理想化包裹圆环示意图



图2-3 目标尖桩示意图

通过一定的竞赛规则，将智能机器人的“竞技”属性进行量化，既可以让我们更加直观地对于机器人的功能进行详细合理地拆分，也可以利用竞技地特性不断推进机器人的改进与发展。

# 3 机械结构部分创新

## 3.1 结构设计

### 3.1.1 底盘设计

①　驱动体系：

我们采用了一种较为独特的电机驱动系统，单边配备四个VEX V5 smart motor电机，这种配置不仅提供了**卓越的动力输出**，还通过精密的齿轮传动系统实现了动力的有效传递。齿轮传动系统的设计确保了单侧的前后从动轮能够以高度一致的转速和扭矩协同工作，从而提供了平稳且高效的驱动性能。

图3-1 机器人电机驱动系统

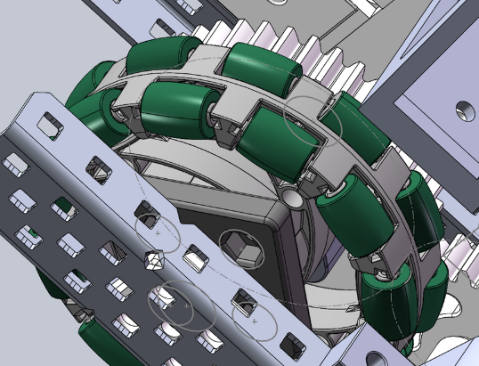
为了使小车能够在多变的环境中灵活地移动，我们特别选用了**万向轮结构**，有助于减少转向时的摩擦力，提高车辆的操控性和机动性，使其能够在多变的环境中保持稳定和高效地运行。

图3-2 万向轮以及轮系示意图

②　气动系统

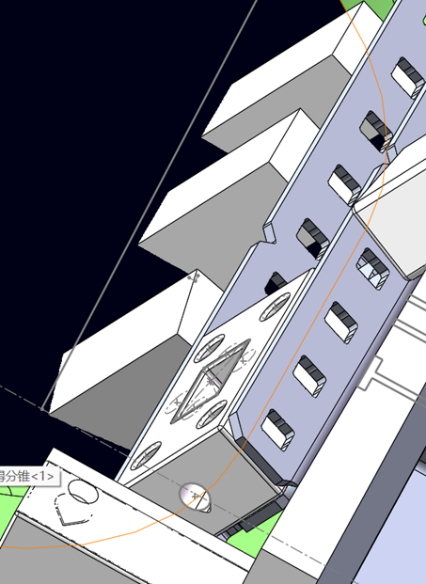
为了优化底盘空间的使用，我们将气泵夹持结构巧妙地布置在底盘的尾部。这一夹持结构设计精巧，能够通过气动驱动实现对正六边形的移动桩基座的夹持，具备**从不同角度**进行**精确夹持的能力**。通过气动系统的精确控制，夹持结构能够实现对移动桩基座的快速夹持和释放，同时保证夹持力的稳定性和可调性。

图3-3 气动系统示意图

③　实用场景：

在自动化生产线上，用于物料的搬运和组装小车若使用万向轮结构，就能让在紧凑的空间内进行多方向移动成为可能，同时在仓库内进行货物的搬运和分类，其气动夹持结构可以稳定地夹持不同形状和尺寸的货物。

### 3.1.2 上层结构设计

为了提高小车的可维护性、便于拆卸和维修，并考虑到未来可能的迭代升级，我们采用了一种创新的**模块化**设计方法。具体来说，我们将集成了吸取、运输和传送带抬升功能的上层总结构与底盘进行了分离建模。这种设计策略不仅优化了空间布局，还显著提升了系统的灵活性和可扩展性.

图3-4 上层结构示意图

①　传送带抬升：同轴传动下的曲柄摇杆机构：

该传送带系统包括一个机架，主动齿轮和从动齿轮分别安装在机架的两端，以及一个由从动轮指向主动轮的传送带。在主动轮的一侧，我们设计了一个抬升装置，该装置通过一个同轴齿轮传动，电机驱动小齿轮将以较高的转速旋转，而大齿轮则以较大的扭矩进行抬升，用以驱动高扭矩低转速需求的传送带。抬升齿轮与传送带之间通过连杆连接，连杆两端分别铰接于传送带侧立面。齿轮系通过连杆驱动升降台上下反复运动，从而实现物品的抬升。

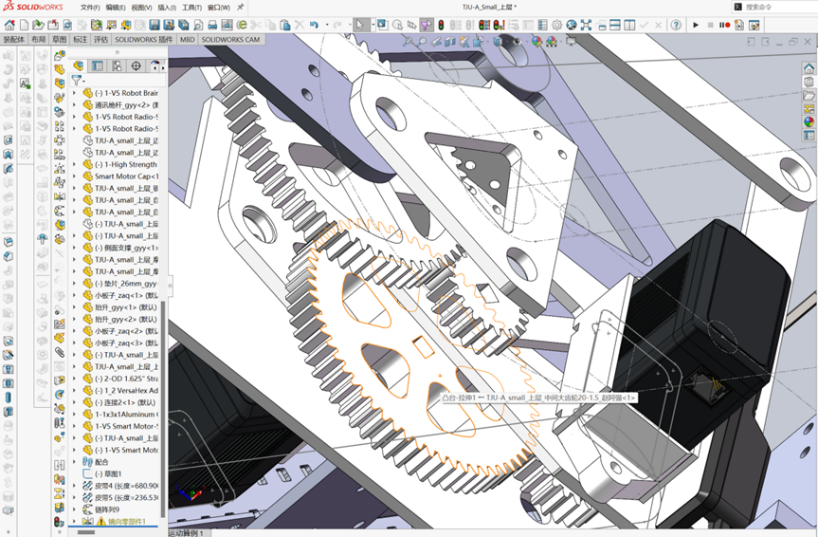


图3-5 抬升轮系示意图

这种设计不仅能够平稳地传送物品，还能通过抬升装置实现物品的垂直移动，为物料搬运提供了额外的灵活性。通过抬升装置利用传送带的主动轮进行驱动，我们的传送带系统降低了能源消耗，使用灵活，并且可以在现有传送带的基础上进行改装。此外，该系统结构简单，重量轻，使用方便，制造成本低，易于维护和升级。

②　链轮传动的吸取结构

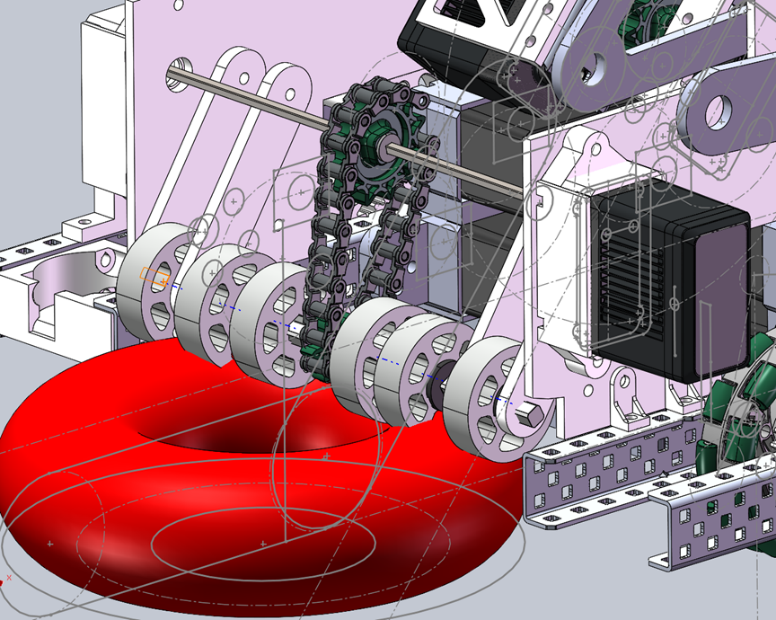
在吸取结构的设计中，巧妙地采用了链轮传动系统驱动并排摩擦轮，利用啮合原理实现动力传递。该系统结构紧凑、传动精确、效率高，且运行稳定。同时为了应对垂直对齐误差，设计了可调节的上下活动结构，通过微调摩擦轮以补偿制造公差、磨损或操作变化，确保最佳接触状态，提升系统稳定性和可靠性。

图4-6 链轮传动设计

③　实际应用

在实际应用中，这种传送带系统可以用于多种场景，包括但不限于生产线的物料输送、仓储物流中的货物搬运等。在物料搬运和自动化物流系统中，传送带的设计对于提升效率和降低成本至关重要。我们设计的传送带系统通过集成抬升功能，实现了物品的垂直移动，从而增强了物流和物料搬运过程中的灵活性和效率

## 3.2 材料选择

在设计机器人小车的结构时，确保其具备足够的刚性和强度是至关重要的。这一要求是为了支撑小车在静态状态下的重量，以及在动态操作过程中可能遇到的各种动态载荷。为了满足这些力学性能的需求，同时考虑到成本效益和实施的可行性，我们选择了一种**综合性能优异**的材料组合方案。

3D打印（3DP）又称增材制造技术（Additive Manufacturing Technologies，AM），是一种依据三维CAD数据通过逐层材料累加的方法制造实体零件的技术。

激光切割是利用经聚焦的高功率密度激光束照射工件，使被照射的材料迅速熔化、汽化、烧蚀或达到燃点，同时借助与光束同轴的高速气流吹除熔融物质，从而实现将工件割开。激光切割属于热切割方法之一。

利用以上两种主要的加工工艺与制造方式，我们完成了搭建智能机器人的基本材料的准备。同时多种材料、工艺的融合也让我们的机器人有了**更好的力学性能**。

具体来说，我们采用了金属、铝方管和铝件作为主要的结构材料，辅以PETG和PLA两种类型的3D打印件。前者金属件以其高强度和良好的刚性而著称，能够有效地承受预期的载荷，而后者3D打印材料不仅提供了**额外的设计灵活性**，还能在不牺牲结构完整性的前提下，实现**成本效益的优化**。

以下是针对底盘结构、传送带结构以及其他辅助固定和承载应用的具体材料选择和设计思路：

1. 底盘结构：

金属材料：底盘的主要承载部分采用高强度金属材料，这种材料能够提供必要的刚性和强度，以支撑整个车辆的重量，并吸收来自地面的冲击。

铝方管和标准铝件：关键的连接点和支撑结构使用铝方管和标准铝件，以确保在保持轻量化的同时，提供承重强度和耐久性。

1. 传送带结构：

PETG材料：在传送带的固定支撑框架以及齿轮部分使用PETG材料3D打印而成，这种材料具有良好的柔韧性、高抗冲击强度和良好的热稳定性，使其在承受重复的物理应力和温度波动时，仍能保持部件的完整性和功能性。

PLA材料：在传送带轻质的导轨和轻载部件使用PLA材料3D打印，这种材料成本较低，且易于加工，适合快速原型制作和成本敏感的应用。

1. 辅助固定和承载应用：

结构加固：在需要额外加固的区域，如连接点和受力集中的部位，使用了铝制部件进行加固，以提高整体结构的承载能力和耐久性。

轻质连接：对于需要频繁拆卸或调整的部件，可以使用PLA或PETG材料制造的轻质连接件，这些部件既方便更新迭代，又能减轻整车重量。

通过这种精心设计的材料组合，机器人小车不仅能够满足承载和稳定性的需求来支持其在各种操作条件下的稳定性和耐用性，还能在成本控制和环境可持续性方面取得平衡。

# 4 格斗机器人AI、程序创新

## 4.1 程序

程序的控制编程主要由VEX V5官方提供的VS Code编译扩展实现，然后通过C++集成编程。

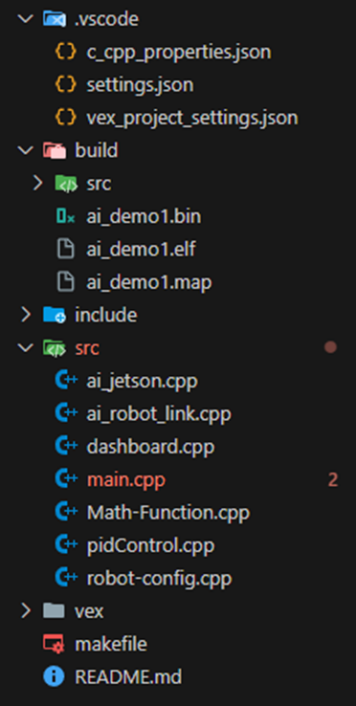
项目集成如下：

图4-1 控制编程介绍

### 4.1.1 底盘控制

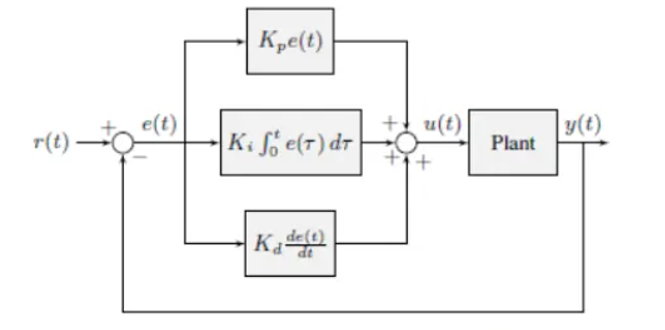
底盘使用了**PID（比例-积分-微分）的闭环控制原理**控制车型走直线，PID（比例-积分-微分）控制是一种常见的控制算法，它广泛应用于工业过程控制、机器人控制、自动驾驶等领域。PID控制的基本原理是根据被控对象的当前状态与设定值之间的差异（即误差）来调整输出信号，使得差异趋近于零，实现稳定精准的自动控制。

图4-2 PID 控制器流程图

PID控制器的工作原理包括传感器检测、PID计算、输出调节和反馈控制四个步骤：

1. 传感器检测：

系统首先通过传感器实时检测被控对象的状态，并将其与设定值进行比较，得到误差。

1. PID计算：

PID控制器根据比例、积分和微分三个部分的权重，将误差转换为输出信号。具体计算方式为：输出信号=Kpe(t)+Ki∫e(t)dt+Kd\*de(t)/dt，其中Kp、Ki和Kd分别为比例、积分和微分系数，e(t)为误差，∫e(t)dt为误差的积分，de(t)/dt为误差的微分。

1. 输出调节：

PID控制器将计算得到的输出信号传递给执行器，执行器根据该信号调节被控对象的状态。这可以是通过改变电流、压力、速度等方式来实现。

1. 反馈控制：

系统不断重复上述步骤，通过不断调整输出信号，使得被控对象的状态逐渐接近设定值。同时，通过传感器不断监测被控对象的实际状态，实现反馈控制，进一步修正输出信号，以实现更加精确的控制。  
 在本机器人的地盘移动函数中也同样使用了pid函数：

图4-3 底盘移动函数示意图

### 4.1.2 功能函数

在机器人的特殊功能中，特定的实现即可。

## 4.2 AI应用

### 4.2.1 基本detection

在基本的objection detection部分，我们使用了**jetson nano b01 基础版的计算版**，搭配D435i，**基于双目的深度相机**，配合上yolo系列训练好的objection detection的pt模型转化为onnx推理模型部署到jetson nano中调用，ONNX是一种开放的神经网络交换格式，它定义了一组与环境、平台均无关的标准格式，用于存储训练好的模型，包括模型的权重、结构信息以及每一层的输入输出等，旨在解决不同框架和硬件平台之间的互操作性问题，使得模型可以在不同的深度学习框架和硬件平台之间自由迁移和部署。

### 4.2.2 yolo

我们自己训练了**2000+**的训练集和**300+**的验证集，来训练yolo模型。

图4-4 yolo环境与训练集、验证集

在100为一个单位的batch训练了300个epoch后，损失函数和map趋近收敛，平均精度**达到83以上**，detection的可视化如下：

图4-5 detection 的可视化

仍然存在的detection问题就是置信度太高，导致模型处理的result返还速度下降。

### 4.2.3 detection的python部署

在借助yolo集成好的库中电泳preprocessyolo和postprocessyolo模型实现数据的接受传递预处理，

然后从onnx中调用get\_engine函数讲推理模型onnx中构造好的参数实现model构造，然后把构造的model交付给dataprocess.py进行进一步使用：

1. Model类
2. get\_engine方法：

这是一个静态方法，用于尝试加载一个已经存在的TensorRT引擎文件。如果文件不存在，它会从ONNX文件构建一个新的引擎，并将其保存到指定的路径。这个方法首先检查引擎文件是否存在，如果存在则加载它；如果不存在，则通过以下步骤构建引擎：

1. 创建一个TensorRT构建器、网络、配置和ONNX解析器。
2. 设置最大工作空间大小和批量大小。
3. 检查ONNX文件是否存在，然后解析它。
4. 设置网络的输入形状。
5. 构建网络并序列化，然后创建并返回引擎。
6. \_\_init\_\_方法：

初始化模型实例，加载TensorRT引擎，创建执行上下文，并分配输入和输出的缓冲区。

1. inference方法：

对给定的图像执行推理，返回检测到的对象的边界框、分数和类别。这个方法首先使用PreprocessYOLO类处理输入图像，然后执行推理，并使用PostprocessYOLO类处理输出，最后绘制边界框并返回处理后的图像和检测结果。

1. draw\_bboxes方法：

这是一个静态方法，用于在原始图像上绘制边界框，并返回修改后的图像。这个方法还会将检测到的对象信息存储在一个列表中。

1. rawDetection类：

这个类用于存储检测到的对象的信息，包括x和y坐标、中心点、宽度、高度、概率和类别ID。

### 4.2.4 数据处理

在dataprocessing.py中，我们调用了you only look once （YOLO）模型的经典思想，对象框提议，然后实现一个基础的**非极大抑制（NMS）**，NMS是一种后处理技术，用于在目标检测过程中去除多余的边界框，从而提高检测的准确性。在目标检测中，经常会出现多个重叠的候选边界框，它们可能对应于同一个目标。NMS通过保留最佳的边界框并抑制其他重叠框来解决这个问题。考虑到本场景中会出现多个目标叠加在一起的情况，我们把NMS后处理放在首要的位置。

1. load\_label\_categories 函数

目的：从指定的文本文件（labels.txt）中加载类别名称。

参数：label\_file\_path - 类别名称文件的路径。

返回值：包含所有类别名称的列表。

1. PreprocessYOLO 类

目的：处理输入图像，使其适合YOLOv3模型的输入要求。

1. \_\_init\_\_：

初始化类，设置YOLOv3的输入分辨率。

1. process：

处理输入图像，包括加载、调整大小和归一化，然后返回原始图像和处理后的图像。

1. \_load\_and\_resize：

加载图像并调整其大小到指定的输入分辨率。

1. \_shuffle\_and\_normalize：

将图像数据归一化到[0, 1]范围，并调整数据布局以符合模型要求。

1. PostprocessYOLO 类

目的：处理YOLOv3模型的输出，提取检测到的对象的边界框、类别和置信度。

方法：

1. \_\_init\_\_：

初始化类，设置YOLOv3的掩码、锚点、对象阈值、NMS阈值和输入分辨率。

1. process：

处理模型的输出，返回检测到的对象的边界框、类别和置信度。

1. \_reshape\_output：

将模型的输出重新整形为适合进一步处理的形式。

1. \_process\_yolo\_output：

处理所有输出，包括应用NMS算法以减少重叠的边界框。

1. \_process\_feats：

计算每个网格单元中检测到的边界框、置信度和类别概率。

1. \_filter\_boxes：

根据对象阈值过滤边界框。

1. \_nms\_boxes：

应用非最大抑制（NMS）算法，以减少相邻且重叠的边界框数量。

关键点如下：

1. YOLOv8输出处理：

YOLOv8模型输出三个不同尺度的特征图，每个特征图对应不同大小的对象。这些输出需要进一步处理以提取有用的信息（如边界框、类别和置信度）。

1. 非最大抑制（NMS）：

是一种用于减少检测到的边界框数量的技术，特别是当多个边界框重叠时。它保留具有最高置信度的边界框，并删除与之重叠的、置信度较低的边界框。

1. 数据归一化和重新排序：

输入图像需要被调整大小、归一化，并重新排序以符合YOLOv3模型的输入要求。同样，模型的输出也需要被重新整形和处理以提取有用的信息

### 4.2.5 检测可视化

随后我们考虑方便对机器进行调试，实现了**dashboard可视化**。

定义了一个名为 V5WebData 的类，它是一个用于处理与 WebSocket 客户端通信的服务端应用程序。这个类主要用于在一个指定的端口上启动一个 WebSocket 服务器，从而允许客户端通过 WebSocket 连接接收和发送数据。这个服务器主要用于处理和传输与机器人或类似设备相关的实时数据，如图像、位置、检测结果、统计信息等。下面是代码的主要组成部分及其功能的详细解释：

类 Statistics：

功能：用于存储和报告实时统计信息，如帧率（FPS）、推理时间、CPU温度、视频宽度和高度、运行时间以及GPS连接状态。

类 V5WebData；

构造函数：

初始化服务器端口、加载GPS和相机的偏移量、颜色校正值，并创建WebSocket服务器实例。

start 方法：启动WebSocket服务器，并初始化用于存储检测数据、图像和统计信息的变量。

1. \_\_new\_client 方法：当有新客户端连接时调用，打印客户端ID。
2. \_\_client\_left 方法：当客户端断开连接时调用，打印断开连接的客户端ID。
3. \_\_message\_received 方法：处理从客户端接收到的消息。根据消息内容，它可能更新GPS或相机偏移、颜色校正值，或者返回请求的数据（如位置、检测、统计、图像等）。
4. setDetectionData、setColorImage、setDepthImage、setStatistics 方法：这些方法用于更新检测数据、彩色图像、深度图像和统计信息。
5. setGpsOffset、setCameraOffset、setColorCorrection 方法：这些方法用于更新GPS偏移、相机偏移和颜色校正值，并将更新保存到JSON文件中。
6. isConnected 方法：检查是否有客户端连接。
7. stop 方法：优雅地关闭服务器。
8. \_\_del\_\_ 方法：析构函数，当对象被删除时调用stop方法。

类 Offset、GPSOffset、CameraOffset：  
功能：这些类用于表示不同类型的偏移量（GPS偏移和相机偏移）。GPSOffset和CameraOffset继承自Offset类，并添加了特定的属性（如CameraOffset中的elevation\_offset）。这些类还提供了从JSON文件加载和保存偏移量的方法。  
 类 ColorCorrection：  
功能：用于存储和处理颜色校正值（HSV值）。提供了从JSON文件加载和保存颜色校正值的方法。

辅助方法：

1. \_\_getStatsElement、\_\_getPositionElement、\_\_getDetectionElement、\_\_getColorElement、\_\_getDepthElement：

这些方法用于获取当前统计信息、位置、检测数据、彩色图像和深度图像的元素，以便通过WebSocket发送给客户端。

1. convert\_numpy\_to\_list：

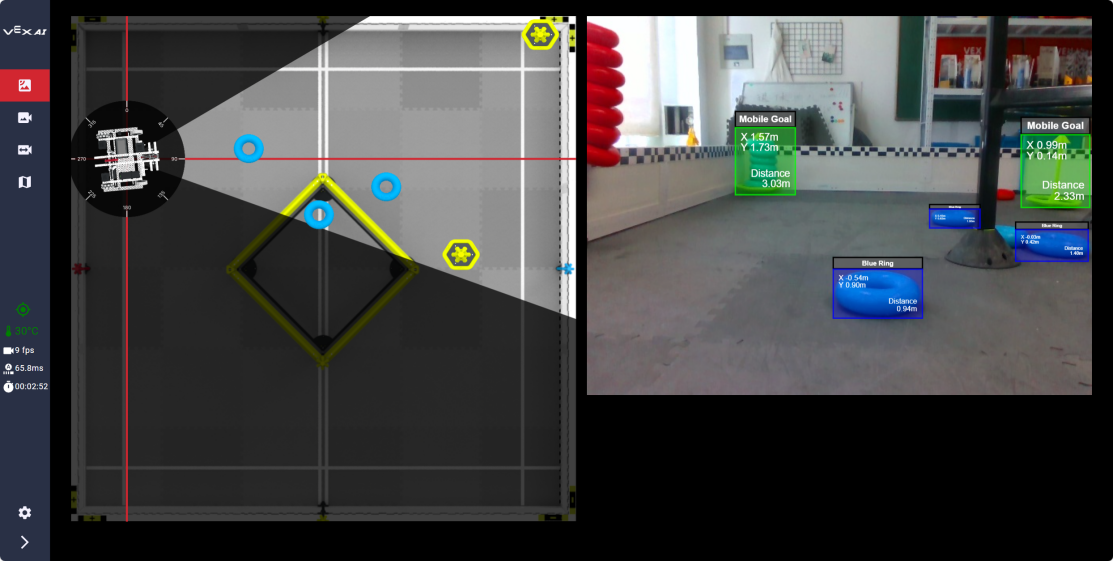
一个辅助函数，用于递归地将NumPy数组转换为列表，以便可以将数据作为JSON发送。

图4-6 可视化效果

### 4.2.6 主控通信

在jetson nano 主板上处理完检测到的目标消息后，要和模拟环境中规定使用的v5主控实现通。

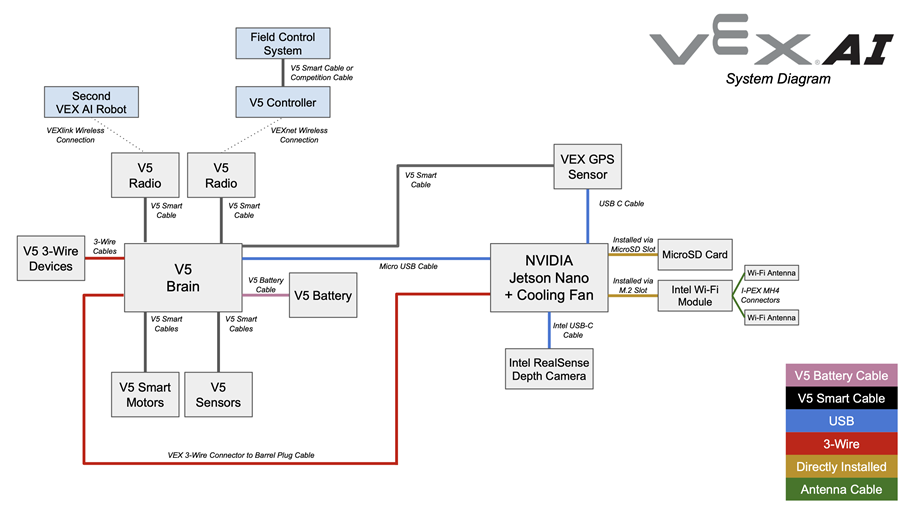
在硬件连接中，使用有线通信将jetson nano和v5主板连接实现通信，这样通信的效率能够满足detection的处理需要。

图4-7 主控通讯结构图

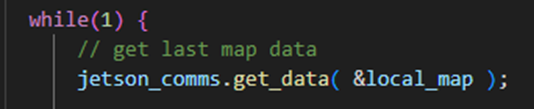
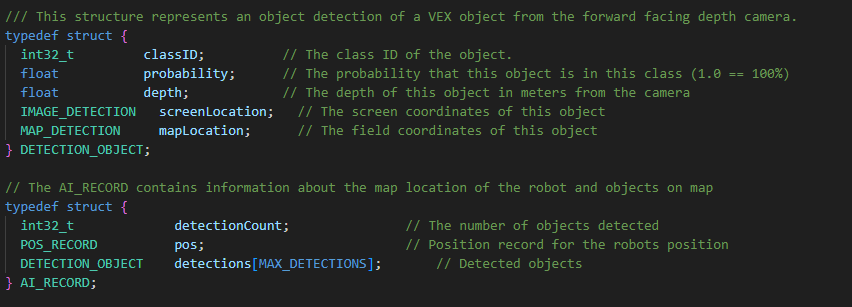
在v5主板上实现控制，此时必须使用cpp编程，好在有request map函数对上文提到的检测到的数据 rawDetection类 去保存成为本地的一个local map变量。

图4-8 函数变量

local\_map的变量类型是AI\_RECORD 里面包含了检测的物体的类型，位置如下：

图4-9 函数位置

### 4.2.7 决策

此时基础控制和感知都实现了，那么可以考虑一下在主板上执行决策问题了。  
 我们模拟实现的是一个基于VEX的搬运问题。那么我们考虑这是一个**带有回归的基于旅行商问题的算法优化**。我们暂定场上需要搬运的物体数量级是1e2级别的，复杂度分析来看时间复杂度就是**O(2^n\*n^2)**，也就是接近每次处理1e7级别的数据，然而Jetson Nano的算力达到1e10级别，每秒钟能够实现1000次的该运算，因此算力不是问题。  
图4-10 Jeston Nano性能图

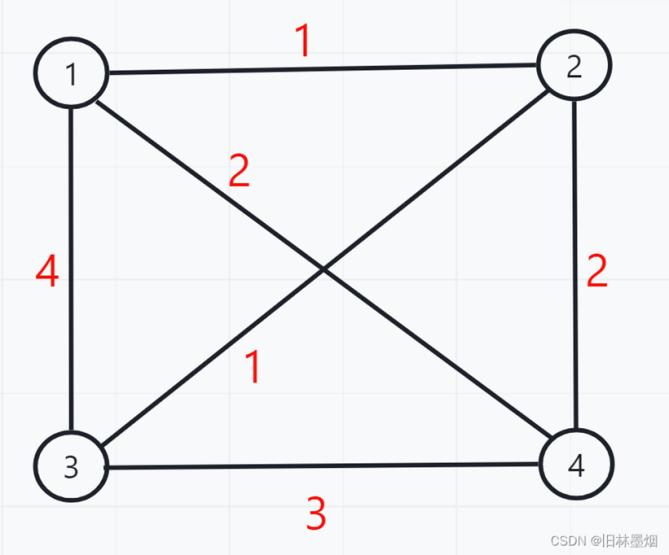
那么在决策的算法可以暂定为**TSP问题**，接下来我们简单看看初始的TSP问题：假设有一个旅行商人要拜访N个城市，他必须选择所要走的路径，路径的限制是每个城市只能拜访一次，而且最后要回到原来出发的城市。路径的选择目标是要求得的路径路程为所有路径之中的最小值。TSP问题是一个NPC问题。

图4-11 TSP问题示意图

从自己所在的点开始的最短遍历路径则是1–>4–>3–>2–>1

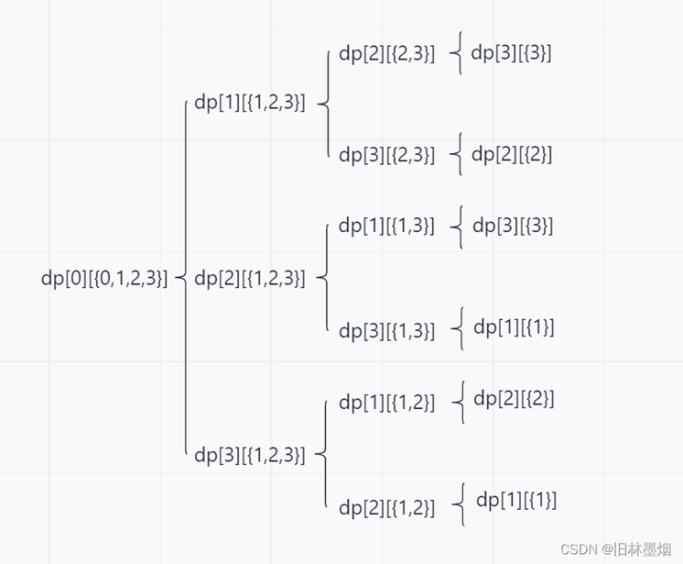
那么这个问题如何扩展为本项目的决策：可以把点换为我们需要去取到的物品，边权转化为我们两点之间的距离（因为距离的比例永远不变，所以可以直接用长度m为单位作为他们的边权w），然后整个图应该是一个完全图（因为你可以从任意一个点出发去到另外一个点去做走）

就拿这个为例子，我们执行一个动态规划：

1. 状态压缩

我们需要表达我们已经走过了哪些点，目前到达了哪里，有什么办法表达出来呢？  
  暴力是万能的，我们可以开一个数组dp[i][j]，代表目前到达了i点，dp[i][j]的值代表j点是否已经走过了，但是这样做的话我们状态转移会变得很麻烦，状态压缩就是它的优化。  
  状态压缩是通过二进制实现的，我们知道int有32位，那么我们可以用第0位代表第0个点的状态，第1位代表第1个点状态…第n位代表第n个点的状态，位的值如果是1的话就代表该点已经走过了，例如17的二进制为0000010001，代表第0个点和第4个点已经走过了  
  那么我们可以开一个数组dp[i][j]，代表目前走到了i点，用j代表已经走过了哪些点，例如：  
dp[0][17]，17的二进制为0000010001,代表目前在第0个点，已经走过第0个点和第4个点。  
dp[4][17]，17的二进制为0000010001,代表目前在第4个点，已经走过第0个点和第4个点。  
  
  我们可以用dp[i][j]的值代表当前这个状态的最小花费，例如dp[0][17]=12，那么就代表到达该状态需要的最小花费是12。

1. 状态转移

 dp的基本思想就是记录某个状态的最优解，再从目前的状态转移到新的状态，从局部最优解转移到全局最优解。  
  我们用数组a[i][j]存储图，那么a[i][j]的值就代表从i点到j点的花费。  
  我们如何求状态dp[0][19]的最优解？  
图4-12 状态最优解

19的二进制是0000010011，因为18的二进制为0000010010，那么dp[0][19]可以由dp[4][18],dp[1][18]转移过来，最小花费是dp[0][19]=min(dp[4][18]+a[4][0],dp[1][18]+a[1][0])。  
  即我们要求大的状态，那么就需要先把小状态最优解求出来。反过来我们求出了所有小状态，那么就可以求出大状态的最优解

{1，2，3}代表第1、2、3个点都已经走过了

可以发现，小状态总是比大状态小的，那么我们可以从0状态枚举到2n-1状态，获取到每个状态的最优解。

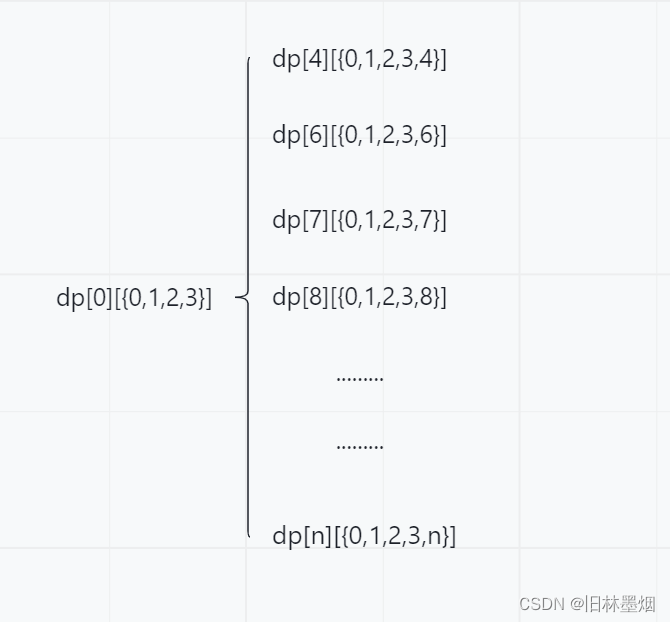
我们还可以反过来想，从小状态去更新大的状态：

图4-13 小状态更新大状态

落实到具体的代码则是先要实现一个从local\_map中获取物体存储在自己的数据结构中的操作：

其中，比对目前数据结构和local\_map中每一个物体位置，差距如果大于diff，那么就判断他为一个新的物体。

这个diff的选择是有意义的，如果太小，那么很容易把有误差的检测同一个物体判断为两个不同的物体，如果太大，那么会导致两个邻近的物体被判断为不存在，那么就漏掉了物体。

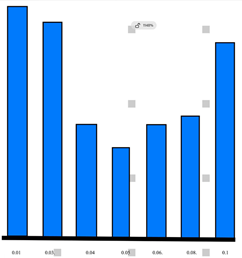
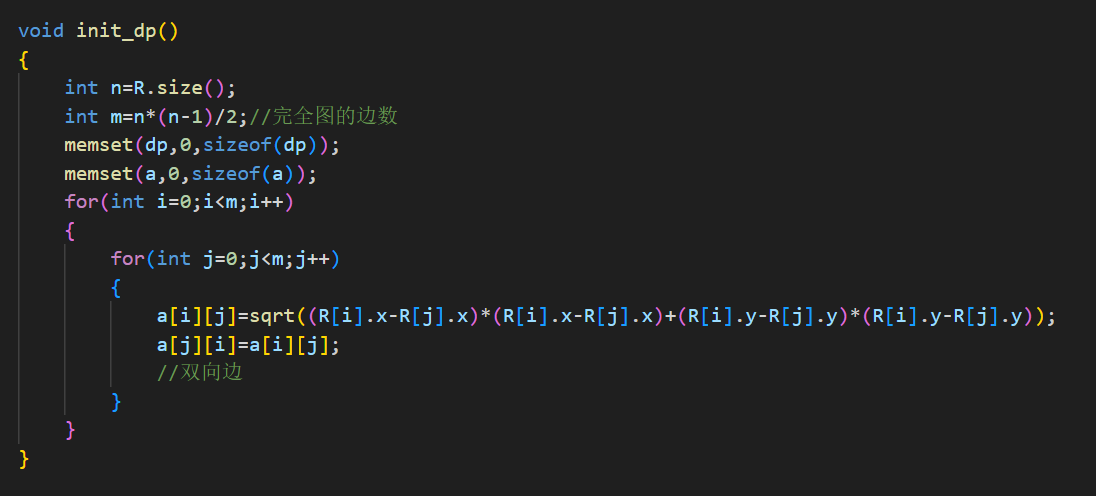
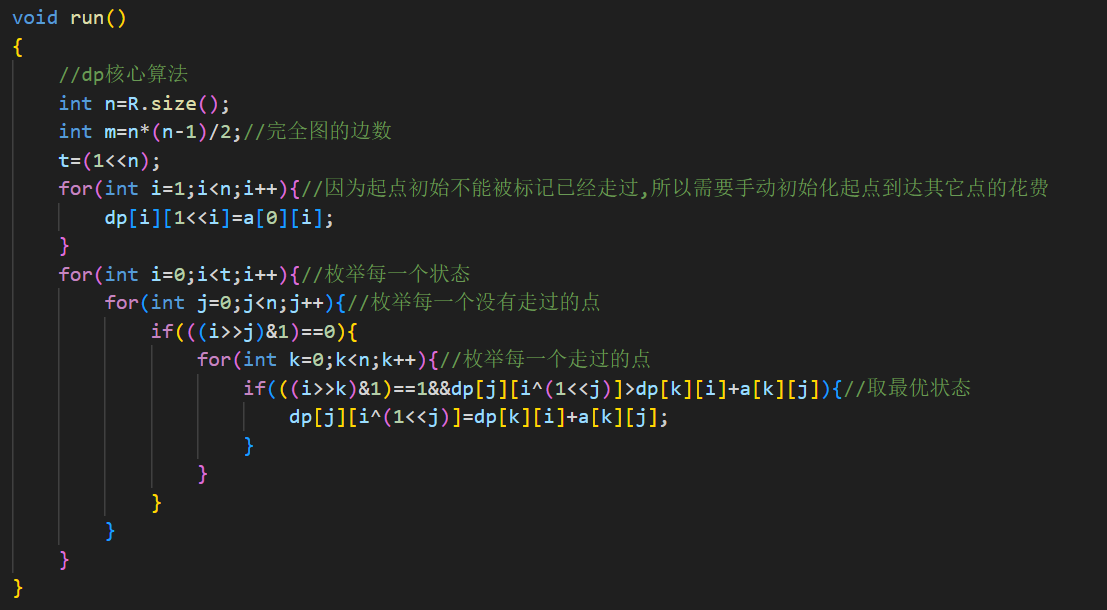
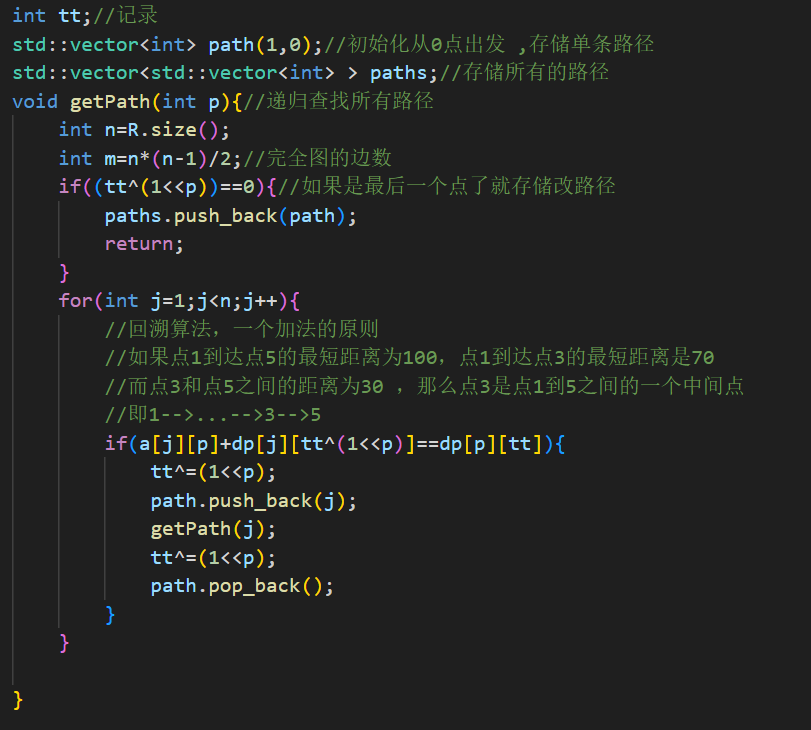
所以我们对diff执行了一个枚举，跑了几次RUN，然后记录漏下的物体数量，效果如下：

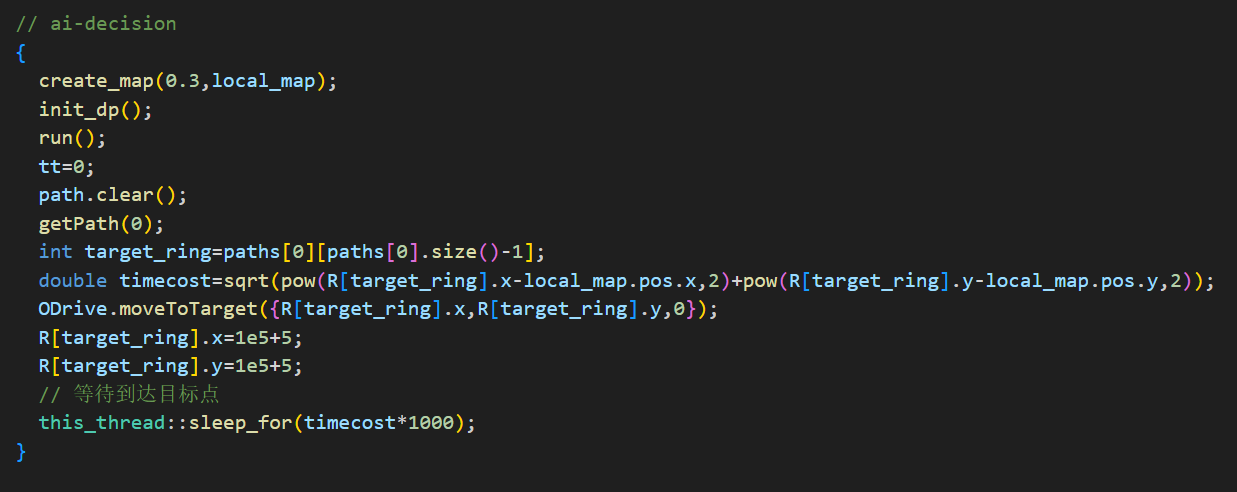
图4-14 效果图

因此最后diff选择了0.05。

然后**初始化所有已经检测到的点**



**注意是双向边：**  
  
  
最后**模拟**之前图示的**动态规划**  
  
     


然后把路线记录在paths中  
注意的是，我们机器人是实时移动更新数据，在while循环中不断获得信息，不断从jetson nano中获得数据，更新自己的数据结构。

注意每次决策执行  
  
    ODrive.moveToTarget({R[target\_ring].x,R[target\_ring].y,0});  
  
之后，要把取到的点直接“抛出地图”，设置成1e5+5；

### 4.2.8 应用

整个机器人实现的内容是一个基础的套取，搬运。提高在于规划，决策，控制。在应用层面，可以进行**模块化设计**：

1.搬运内容：

关于要搬运的内容，在本设计中理想化的设计为了“环”，在实际的应用中，可以根据机械情况和搬运目标做出改变，比如搬运一定的快递件的时候可以设定为“立方体”设计搬运的程序和机械设计。

2.检测内容：

基于yolo实现的目标检测有相当的优越性，它**轻量**、方便训练、部署方式成熟。可以根据需要搬运的内容，构建labelimg训练集训练yolo框架转化为onnx推理模型进行jetson nano的部署。数据量可以不用很大，因此本任务在迁移学习中效果极好。只需要10的3次方级别的数据集就可以训练一个满足基本需求的检测模型

3.决策方式：

具体的决策方式可以考虑具体的场景，在这个设计中没有避障的需求，只有一个遍历全部内容的需要，把全部的环在尽量少的时间内全部抓取，因此我们使用旅行商问题（TSP）的变体，控制好要抓取的内容和起点终点。

如果场景有变，只需要更改对应的决策类型即可。比如说对快递的搬运放取，可以考虑实现一个最短路或者最小生成树的算法。如果带有避障内容，可以考虑加入一个搜索的模块。如果场景实在过于复杂，可以使用基于决策树的决策算法。

我们把决策的内容统一使用了一个model.py执行决策内容，实现了决策的**模块化设计**

综上三点模块化的设计，我们的项目设计可以应用到许多其他的内容。

# 5 总结与未来展望

## 5.1 总结

综上，我们通过相对低成本的VEX V5硬件，轻量化的目标检测平台yolo，合适的小规模算力平台jetson nano，以及模块化设计的决策算法。实现了一个以简单**·**的搬运设计。得益于**模块化的设计**，这个设计在修改到其他应用场景上迁移任务完成效果会相对的好，而且轻量简单，试错成本较低，设备迭代更新就会更快。

与此同时，我们的机械结构设计也饱含创新性，通过多种加工工艺与不同的材料的相结合，充分提高了结合利用的高效性，有效降低智能机器人建造成本；通过不同的机械结构的组合，成为机器人的功能实现载体，与AI功能相辅相成，共同组成了我们的模块化、轻量化设计方案。

## 5.2 未来展望

从目前的社会发展状况来看，人工智能赋能技术路线将会继续不断在未来的洪流当中发挥其不可忽视的作用，因此我们所设计的“基于VEX系统的未来式物流搬运模块化轻量化智能机器人”能够在未来提出一套独特而行之有效的解决方案。

我们预计将在未来继续不断完善其外部功能结构载体，在追求不断轻量化、高效化的过程中探索将理想化模型映射到实际上的方法论。通过利用更多丰富的机械结构与材料组合实现更加便利的搬运功能

同时在AI运用方面我们也有更多的想法与展望：

一方面我们希望能把这一套基于vex的模块化轻量搬运设计应用到更多的场景上，这也是这个项目的初衷。

另一方面，我们希望能优化我们的目标检测过程和硬件之间的通信功能，在实现ai算力平台和控制平台的通信中，我们发线有线通信出现大约10帧左右的卡顿，之后我们希望更新到蓝牙通信解决这个问题。然后我们希望实现一个比较通用的决策方式，去减少这个项目在迁移到其他任务上算法决策的修改成本，强化该项目的轻量化性能和模块化性能。